Acceleration of Optimization-based Structured Sparse Time-Frequency Analysis by ADMM

Keidai ARAI and Kohei YATABE
Tokyo University of Agriculture and Technology, Tokyo, Japan

***Abstract*—Sparse time-frequency (T-F) analysis has been studied to obtain localized T-F representations of a signal. Among various methods, optimization-based methods (e.g., basis pursuit) offer flexibility in designing T-F representations by designing the objective function. In particular, the convex-optimization-based method that imposes a specified structure on the magnitude of a T-F representation has realized T-F analysis tailored for specific applications. However, the conventional method uses the PDS (primal-dual splitting) algorithm, which is known to require many iterations in some cases. In this paper, we propose applying ADMM (alternating direction method of multipliers) with acceleration techniques to reduce the number of iterations required for obtaining structured sparse T-F representations. Experiments show that the proposed algorithm can obtain the T-F representation much faster than the conventional PDS algorithm.**

***Index Terms*—Structured sparsity, convex optimization, perspective function, Nesterov acceleration, optimal step-size.**

I. Introduction

Time-frequency (T-F) analysis is an indispensable tool in acoustic signal processing [1], [2]. Among various approaches for T-F analysis, the discrete Gabor transform (DGT) is widely used owing to its simplicity and good performance. Let DGT of a signal $\mathbf{s} \in \mathbb{R}^L$ using a window $\mathbf{w} \in \mathbb{C}^L$ be defined as

$$x[m,n] = \sum_{l=0}^{L-1} s[l]\, \overline{w[l-an]}\, \mathrm{e}^{-2\pi \mathrm{i} ml/M}, \tag{1}$$

where $L \in \mathbb{N}$ is the signal length, $n = 0, \ldots, N-1$ and $m = 0, \ldots, M-1$ are the time and frequency indices, respectively, $N, M \in \mathbb{N}$ are the number of time and frequency bins, respectively, and $a \in \mathbb{N}$ is the time step of DGT. In this paper, we shortly write Eq. (1) as $\mathbf{x} = \mathbf{G_w s}$, where $\mathbf{x}$ is the vectorized version of the left-hand side of Eq. (1), and $G_w[m+nM, l] = w[l-an]\, \mathrm{e}^{-2\pi \mathrm{i} ml/M}$.

A T-F representation obtained by DGT exhibits spread due to the windowing. To suppress spread and obtain a localized representation, sparse-optimization-based methods have been studied. These methods can promote a certain structure in T-F representations by choosing a proper prior. The simplest convex method is minimization of the ℓ_1-norm of the representation under the perfect reconstruction constraint [3]–[6]:

$$\min_{\mathbf{x}} \quad \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \mathbf{G}_{\mathbf{w}}^{\dagger}\mathbf{x} = \mathbf{s}, \tag{2}$$

where $\mathbf{G}_{\mathbf{w}}^{\dagger}$ is the pseudo-inverse of $\mathbf{G_w}$. There are many other functions available for inducing desired structures [7]–[15].

In acoustic applications, it is often assumed that the magnitude of a T-F representation exhibits a specific structure across T-F bins, such as piecewise smoothness or being approximately low-rank [16], [17]. When imposing such structures on the magnitude, direct formulation results in a nonconvex problem, so optimization algorithms may get stuck in a local minimum. To resolve this issue, a convex optimization framework has been proposed for structured sparse T-F representations [18]. However, this framework uses the PDS (primal-dual splitting) algorithm to solve the optimization problem, which is known to require many iterations in some cases [19].

In this paper, we propose an ADMM (the alternating direction method of multipliers) algorithm [20] for structured sparse T-F analysis and apply some acceleration techniques [21]–[23] to it. As indicated by the experimental results, the proposed algorithm can obtain structured sparse T-F representations much faster than the conventional PDS algorithm. Our main contributions are summarized as follows: (i) deriving an ADMM algorithm tailored for structured sparse T-F analysis; (ii) incorporating acceleration techniques into the ADMM algorithm; and (iii) evaluating effectiveness of the proposed algorithms by qualitative and quantitative experiments.

II. Structured Sparse T-F Analysis via Convex Optimization

The magnitude spectrogram of an acoustic signal has specific structure across T-F bins. For example, most acoustic signals change continuously over time, resulting in piecewise smoothness of the magnitude spectrograms. Moreover, for harmonic instruments, they often exhibit an approximate low-rank property. Given these properties, it is desirable to obtain sparse T-F representations that reflect these structures.

To achieve this, a convex optimization framework has been proposed in [18]. It realizes structured sparse T-F representations by solving the following convex optimization problem:

$$\min_{(\mathbf{x},\boldsymbol{\sigma})} \; \varphi(\mathbf{x},\boldsymbol{\sigma}) + \lambda\psi(\mathbf{B}\boldsymbol{\sigma}) + \iota_{\mathcal{C}}(\mathbf{x}), \tag{3}$$

where $\psi : \mathbb{R}^J \to \mathbb{R} \cup \{+\infty\}$ is a proper lower-semicontinuous convex function, $\mathbf{B} \in \mathbb{C}^{J \times MN}$ is a matrix, $\lambda > 0$ is a regularization parameter, $\iota_{\mathcal{C}}$ is the indicator function (i.e., $\iota_{\mathcal{C}}(\mathbf{x}) = 0$ if $\mathbf{x} \in \mathcal{C}$; $\iota_{\mathcal{C}}(\mathbf{x}) = +\infty$ otherwise) of the set $\mathcal{C} = \{\mathbf{x} \in \mathbb{C}^{MN} \mid \mathbf{G}_{\mathbf{w}}^{\dagger}\mathbf{x} = \mathbf{s}\}$ that satisfies the perfect reconstruction condition of the signal. φ is a convex function defined for a pair $(\mathbf{x},\boldsymbol{\sigma}) \in \mathbb{C}^{MN} \times \mathbb{R}^{MN}$ given by

$$\varphi(\mathbf{x},\boldsymbol{\sigma}) = \sum_{k=1}^{MN} \phi(x_k, \sigma_k), \tag{4}$$

and $\phi : \mathbb{C} \times \mathbb{R} \to \mathbb{R}_+ \cup \{+\infty\}$ is the perspective function of $(|\cdot|^2 + 1)/2$ [24], [25] defined by

$$\phi(x_k, \sigma_k) = \begin{cases} \frac{|x_k|^2}{2\sigma_k} + \frac{\sigma_k}{2} & (\sigma_k > 0), \\ 0 & (x_k = 0 \text{ and } \sigma_k = 0), \\ +\infty & (\text{otherwise}). \end{cases} \tag{5}$$

By selecting ψ and $\mathbf{B}$, the structure induced by $\psi \circ \mathbf{B}$ is reflected in the magnitude. Some examples are given below.

Example 1. Structures induced by $\psi \circ \boldsymbol{B}$ in Problem (3).

1) *Total variation:* Let $\mathbf{D}_{\mathrm{2D}}$ be the two-dimensional difference operator and $\|\cdot\|_{2,1}$ be the mixed $\ell_{2,1}$-norm. By setting $\psi = \|\cdot\|_{2,1}$ and $\mathbf{B} = \mathbf{D}_{\mathrm{2D}}$, $\psi \circ \mathbf{B} = \|\mathbf{D}_{\mathrm{2D}}(\cdot)\|_{2,1}$ represents total variation, inducing piecewise smoothness of the magnitude [26].
2) *Low-rank:* Setting $\psi = \|\cdot\|_*$ and $\mathbf{B} = \mathbf{I}$ induces an approximate low-rank property [27] of the magnitude, where $\mathbf{I}$ denotes the identity operator and the nuclear norm $\|\cdot\|_*$ treats the vector $\boldsymbol{\sigma} \in \mathbb{R}^{MN}$ as an $M \times N$ matrix.
3) *Harmonic enhancement:* Setting $\psi = \|\cdot\|_{2,1}$ and $\mathbf{B} = \mathbf{F}\mathbf{D}_{\mathrm{2D}}$ enhances the harmonic structure of the magnitude [28], where $\mathbf{F}$ denotes the unitary discrete Fourier transform (DFT) in the frequency direction.

III. Proposed Method

Conventionally, Problem (3) was solved by the PDS algorithm (see [18]), which often requires a large number of iterations. Here, we propose the standard ADMM algorithm [20], and then introduce its acceleration in the next section.

A. Derivation of ADMM Algorithm

Using auxiliary variables, Problem (3) can be rewritten as

$$\begin{aligned} \min_{(\mathbf{x}, \boldsymbol{\sigma}, \boldsymbol{\chi}, \boldsymbol{\eta}, \mathbf{z}, \boldsymbol{\xi})} \quad & \varphi(\boldsymbol{\chi}, \boldsymbol{\eta}) + \lambda\psi(\mathbf{z}) + \iota_{\mathcal{C}}(\boldsymbol{\xi}) \\ \text{s.t.} \quad & \boldsymbol{\chi} = \mathbf{x}, \ \boldsymbol{\eta} = \boldsymbol{\sigma}, \ \mathbf{z} = \mathbf{B}\boldsymbol{\sigma}, \ \boldsymbol{\xi} = \mathbf{x}, \end{aligned} \tag{6}$$

where $\mathbf{x}, \boldsymbol{\chi}, \boldsymbol{\xi} \in \mathbb{C}^{MN}$, $\boldsymbol{\sigma}, \boldsymbol{\eta} \in \mathbb{R}^{MN}$, and $\mathbf{z} \in \mathbb{C}^J$. Let

$$\mathbf{A} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{B}^\mathsf{T} & \mathbf{I} \end{bmatrix}^\mathsf{T}, \ \boldsymbol{\nu} = \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\sigma} \end{bmatrix}, \ \boldsymbol{\omega} = \begin{bmatrix} \boldsymbol{\chi}^\mathsf{T} \ \boldsymbol{\xi}^\mathsf{T} \ \mathbf{z}^\mathsf{T} \ \boldsymbol{\eta}^\mathsf{T} \end{bmatrix}^\mathsf{T},$$

where $\mathbf{O}$ is a zero matrix with the proper size, and functions f, g be defined by $f(\boldsymbol{\nu}) = 0$, $g(\boldsymbol{\omega}) = \varphi(\boldsymbol{\chi}, \boldsymbol{\eta}) + \lambda\psi(\mathbf{z}) + \iota_{\mathcal{C}}(\boldsymbol{\xi})$. Then, Problem (6) can be further rewritten as

$$\min_{(\boldsymbol{\nu}, \boldsymbol{\omega})} \quad f(\boldsymbol{\nu}) + g(\boldsymbol{\omega}) \quad \text{s.t.} \quad \mathbf{A}\boldsymbol{\nu} = \boldsymbol{\omega}. \tag{7}$$

Applying ADMM to Problem (7) provides Alg. 1, where the proximity operator $\mathrm{prox}_{\gamma\varphi}$ of $\gamma\varphi$ (with $\gamma > 0$) and the projection $P_{\mathcal{C}}$ are given as follows.

The proximity operator of $\gamma\varphi$ can be computed as

$$\mathrm{prox}_{\gamma\varphi}(\mathbf{x}, \boldsymbol{\sigma}) = (\mathrm{prox}_{\gamma\phi}(x_k, \sigma_k))_{k=1}^{MN}, \tag{8}$$

where the proximity operator for each entry is given as in [25]:

$$\mathrm{prox}_{\gamma\phi}(x_k, \sigma_k) = \begin{cases} (0, 0) & (2\gamma\sigma_k + |x_k|^2 \leq \gamma^2), \\ \left(0, \ \sigma_k - \frac{\gamma}{2}\right) & (x_k = 0 \text{ and } 2\sigma_k > \gamma), \\ \left(x_k - \gamma s \frac{x_k}{|x_k|}, \ \sigma_k + \gamma \frac{s^2 - 1}{2}\right) & (\text{otherwise}), \end{cases} \tag{9}$$

and $s > 0$ is the unique positive root of the cubic equation $s^3 + \left(\frac{2}{\tau}\sigma_k + 1\right) s - \frac{2}{\tau}|x_k| = 0$ that can be solved as in [15]:

$$s = \begin{cases} \sqrt[3]{-\frac{q}{2} + \sqrt{-r}} + \sqrt[3]{-\frac{q}{2} - \sqrt{-r}} & (r < 0), \\ 2\sqrt[3]{-\frac{q}{2}} & (r = 0), \\ 2\sqrt[6]{\frac{q^2}{4} + r} \cos\left(\frac{\arctan\left(-2\sqrt{r}/q\right)}{3}\right) & (r > 0), \end{cases} \tag{10}$$

with $p = \frac{2}{\gamma}\sigma_k + 1$, $q = -\frac{2}{\gamma}|x_k|$, and $r = -\frac{p^3}{27} - \frac{q^2}{4}$.

Algorithm 1 ADMM algorithm for Problem (3)

```
Input: $\mathbf{x}^{[0]}, \boldsymbol{\sigma}^{[0]}, \boldsymbol{\chi}^{[0]}, \boldsymbol{\xi}^{[0]}, \mathbf{z}^{[0]}, \boldsymbol{\eta}^{[0]}, \mathbf{y}_{\boldsymbol{\chi}}^{[0]}, \mathbf{y}_{\boldsymbol{\xi}}^{[0]}, \mathbf{y}_{\mathbf{z}}^{[0]}, \mathbf{y}_{\boldsymbol{\eta}}^{[0]}, \lambda > 0, \rho > 0$
  for $i = 0, 1, 2, \ldots$ do
    $\mathbf{x}^{[i+1]} = ((\boldsymbol{\chi}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\chi}}^{[i]}) + (\boldsymbol{\xi}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\xi}}^{[i]}))/2$
    $\boldsymbol{\sigma}^{[i+1]} = (\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I})^{-1}((\boldsymbol{\eta}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\eta}}^{[i]}) + \mathbf{B}^\mathsf{H}(\mathbf{z}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\mathbf{z}}^{[i]}))$
    $(\boldsymbol{\chi}^{[i+1]}, \boldsymbol{\eta}^{[i+1]}) = \mathrm{prox}_{\varphi/\rho}(\mathbf{x}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\chi}}^{[i]}, \boldsymbol{\sigma}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\eta}}^{[i]})$
    $\boldsymbol{\xi}^{[i+1]} = P_{\mathcal{C}}(\mathbf{x}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\xi}}^{[i]})$
    $\mathbf{z}^{[i+1]} = \mathrm{prox}_{(\lambda/\rho)\psi}(\mathbf{B}\boldsymbol{\sigma}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\mathbf{z}}^{[i]})$
    $\mathbf{y}_{\boldsymbol{\chi}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\chi}}^{[i]} + \rho(\mathbf{x}^{[i+1]} - \boldsymbol{\chi}^{[i+1]})$
    $\mathbf{y}_{\boldsymbol{\xi}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\xi}}^{[i]} + \rho(\mathbf{x}^{[i+1]} - \boldsymbol{\xi}^{[i+1]})$
    $\mathbf{y}_{\mathbf{z}}^{[i+1]} = \mathbf{y}_{\mathbf{z}}^{[i]} + \rho(\mathbf{B}\boldsymbol{\sigma}^{[i+1]} - \mathbf{z}^{[i+1]})$
    $\mathbf{y}_{\boldsymbol{\eta}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\eta}}^{[i]} + \rho(\boldsymbol{\sigma}^{[i+1]} - \boldsymbol{\eta}^{[i+1]})$
  end for
```

The projection $P_{\mathcal{C}}$ onto the set $\mathcal{C}$ is given as in [29]:

$$P_{\mathcal{C}}(\mathbf{x}) = \mathbf{x} - \mathbf{G}_{\mathbf{w}}(\mathbf{G}_{\mathbf{w}}^\dagger \mathbf{x} - \mathbf{s}). \tag{11}$$

Note that the proximity operator of $(\lambda/\rho)\psi$ depends on the choice of the function ψ.

B. Efficient Computation of Inverse Operation

In Alg. 1, $(\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I})^{-1}$ appears in the $\boldsymbol{\sigma}$-update. Since $\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I} \in \mathbb{C}^{MN \times MN}$, the complexity of direct computation is $O(M^3N^3)$, which is inefficient. Here, we show that it can be efficiently computed for each matrix $\mathbf{B}$ in *Example 1*.

First, setting $\mathbf{B} = \mathbf{I}$ gives $(\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I})^{-1} = \mathbf{I}/2$. Next, when $\mathbf{B} = \mathbf{D}_{\mathrm{2D}}$, given that spectrograms have periodic boundary condition, the one-dimensional difference matrix $\mathbf{D} \in \mathbb{R}^{M \times M}$ is defined as a circulant matrix. Then, the two-dimensional difference matrix $\mathbf{D}_{\mathrm{2D}} \in \mathbb{R}^{2MN \times MN}$ given by $\mathbf{D}_{\mathrm{2D}} = [\mathbf{D}_\mathrm{v}^\mathsf{T} \ \mathbf{D}_\mathrm{h}^\mathsf{T}]^\mathsf{T}$ with

$$\mathbf{D}_\mathrm{v} = \begin{bmatrix} \mathbf{D} & & & \\ & \mathbf{D} & & \\ & & \ddots & \\ & & & \mathbf{D} \end{bmatrix}, \quad \mathbf{D}_\mathrm{h} = \begin{bmatrix} -\mathbf{I} & \mathbf{I} & & \\ & \ddots & \ddots & \\ & & -\mathbf{I} & \mathbf{I} \\ \mathbf{I} & & & -\mathbf{I} \end{bmatrix}$$

Algorithm 2 ADMM w/ Nesterov acceleration for Problem (3)

```
Input: $\mathbf{x}^{[0]}, \boldsymbol{\sigma}^{[0]}, \boldsymbol{\chi}^{[0]}, \boldsymbol{\xi}^{[0]}, \mathbf{z}^{[0]}, \boldsymbol{\eta}^{[0]}, \mathbf{y}_{\boldsymbol{\chi}}^{[0]}, \mathbf{y}_{\boldsymbol{\xi}}^{[0]}, \mathbf{y}_{\mathbf{z}}^{[0]}, \mathbf{y}_{\boldsymbol{\eta}}^{[0]}, \lambda > 0, \rho > 0,$
       $\beta^{[0]} = 1,\ 0 < \epsilon < 1,\ r^{[0]} = +\infty$
Note: We define $\boldsymbol{\omega}^{[i]} = \left[(\boldsymbol{\chi}^{[i]})^\mathsf{T}, (\boldsymbol{\xi}^{[i]})^\mathsf{T}, (\mathbf{z}^{[i]})^\mathsf{T}, (\boldsymbol{\eta}^{[i]})^\mathsf{T}\right]^\mathsf{T}$ and $\mathbf{y}^{[i]} =$
      $\left[(\mathbf{y}_{\boldsymbol{\chi}}^{[i]})^\mathsf{T}, (\mathbf{y}_{\boldsymbol{\xi}}^{[i]})^\mathsf{T}, (\mathbf{y}_{\mathbf{z}}^{[i]})^\mathsf{T}, (\mathbf{y}_{\boldsymbol{\eta}}^{[i]})^\mathsf{T}\right]^\mathsf{T}$ in this algorithm below.
  for $i = 0, 1, 2, \ldots$ do
    $\mathbf{x}^{[i+1]} = \big((\boldsymbol{\chi}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\chi}}^{[i]}) + (\boldsymbol{\xi}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\xi}}^{[i]})\big)/2$
    $\boldsymbol{\sigma}^{[i+1]} = (\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I})^{-1}\big((\boldsymbol{\eta}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\eta}}^{[i]}) + \mathbf{B}^\mathsf{H}(\mathbf{z}^{[i]} - \frac{1}{\rho}\mathbf{y}_{\mathbf{z}}^{[i]})\big)$
    $(\boldsymbol{\chi}^{[i+1]}, \boldsymbol{\eta}^{[i+1]}) = \mathrm{prox}_{\varphi/\rho}\big(\mathbf{x}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\chi}}^{[i]}, \boldsymbol{\sigma}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\eta}}^{[i]}\big)$
    $\boldsymbol{\xi}^{[i+1]} = P_{\mathcal{C}}\big(\mathbf{x}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\boldsymbol{\xi}}^{[i]}\big)$
    $\mathbf{z}^{[i+1]} = \mathrm{prox}_{(\lambda/\rho)\psi}\big(\mathbf{B}\boldsymbol{\sigma}^{[i+1]} + \frac{1}{\rho}\mathbf{y}_{\mathbf{z}}^{[i]}\big)$
    $\mathbf{y}_{\boldsymbol{\chi}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\chi}}^{[i]} + \rho(\mathbf{x}^{[i+1]} - \boldsymbol{\chi}^{[i+1]})$
    $\mathbf{y}_{\boldsymbol{\xi}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\xi}}^{[i]} + \rho(\mathbf{x}^{[i+1]} - \boldsymbol{\xi}^{[i+1]})$
    $\mathbf{y}_{\mathbf{z}}^{[i+1]} = \mathbf{y}_{\mathbf{z}}^{[i]} + \rho(\mathbf{B}\boldsymbol{\sigma}^{[i+1]} - \mathbf{z}^{[i+1]})$
    $\mathbf{y}_{\boldsymbol{\eta}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\eta}}^{[i]} + \rho(\boldsymbol{\sigma}^{[i+1]} - \boldsymbol{\eta}^{[i+1]})$
    $\beta^{[i+1]} = (1 + \sqrt{1 + 4\beta^{[i]}})/2$
    $\alpha^{[i+1]} = (\beta^{[i]} - 1)/\beta^{[i+1]}$
    $\hat{\boldsymbol{\omega}}^{[i+1]} = \boldsymbol{\omega}^{[i+1]} + \alpha^{[i+1]}(\boldsymbol{\omega}^{[i+1]} - \boldsymbol{\omega}^{[i]})$
    $\hat{\mathbf{y}}^{[i+1]} = \mathbf{y}^{[i+1]} + \alpha^{[i+1]}(\mathbf{y}^{[i+1]} - \mathbf{y}^{[i]})$
    $r^{[i+1]} = \rho\|\hat{\boldsymbol{\omega}}^{[i+1]} - \boldsymbol{\omega}^{[i]}\|_2^2 + \|\hat{\mathbf{y}}^{[i+1]} - \mathbf{y}^{[i]}\|_2^2/\rho$
    if $r^{[i+1]} < \epsilon^{i+1}r^{[0]}$ then
      $(\boldsymbol{\omega}^{[i+1]}, \mathbf{y}^{[i+1]}) = (\hat{\boldsymbol{\omega}}^{[i+1]}, \hat{\mathbf{y}}^{[i+1]})$
    end if
  end for
```

makes $\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I} = \mathbf{D}_{2\mathrm{D}}^\mathsf{H}\mathbf{D}_{2\mathrm{D}} + \mathbf{I}$, which can be written as

$$\mathbf{D}_{2\mathrm{D}}^\mathsf{H}\mathbf{D}_{2\mathrm{D}} + \mathbf{I} = \begin{bmatrix} \mathbf{D}^\mathsf{H}\mathbf{D} + 3\mathbf{I} & -\mathbf{I} & & -\mathbf{I} \\ -\mathbf{I} & \ddots & \ddots & \\ & \ddots & \mathbf{D}^\mathsf{H}\mathbf{D} + 3\mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & & -\mathbf{I} & \mathbf{D}^\mathsf{H}\mathbf{D} + 3\mathbf{I} \end{bmatrix}. \qquad (12)$$

This is invertible from its positive definiteness. Since $\mathbf{D}_{2\mathrm{D}}^\mathsf{H}\mathbf{D}_{2\mathrm{D}} + \mathbf{I}$ is an $N \times N$ block circulant matrix and each block is an $M \times M$ circulant matrix, its inverse can be computed by the two-dimensional fast Fourier transform with the complexity of $O(MN \log MN)$ [30]. Lastly, for the case where $\mathbf{B} = \mathbf{F}\mathbf{D}_{2\mathrm{D}}$, since $\mathbf{F}$ is unitary, $\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I} = \mathbf{D}_{2\mathrm{D}}^\mathsf{H}\mathbf{F}^\mathsf{H}\mathbf{F}\mathbf{D}_{2\mathrm{D}} + \mathbf{I} = \mathbf{D}_{2\mathrm{D}}^\mathsf{H}\mathbf{D}_{2\mathrm{D}} + \mathbf{I}$ holds, which reduces to the case where $\mathbf{B} = \mathbf{D}_{2\mathrm{D}}$. Overall, the inverse operation in Alg. 1 can be efficiently computed for all cases in *Example 1*.

IV. Acceleration of ADMM Algorithm

In this section, we incorporate two acceleration techniques for ADMM to speed up the algorithm. Specifically, we attempt Nesterov acceleration [21], [22] and an optimal step-size selection [23], both of which preserve convergence of ADMM.

A. Nesterov Acceleration

Nesterov acceleration has been incorporated into ADMM [21], [22]. The ADMM algorithm with Nesterov acceleration for solving Problem (3) is provided in Alg. 2. This involves computation of the combined residual $r^{[i+1]}$ based on the primal residual $\|\hat{\boldsymbol{\omega}}^{[i+1]} - \boldsymbol{\omega}^{[i]}\|_2$ and the dual residual $\|\hat{\mathbf{y}}^{[i+1]} - \mathbf{y}^{[i]}\|_2$ when extrapolating variables, and the acceleration is performed in such a way that the combined residual decreases. If the combined residual is less than the threshold $\epsilon^{i+1}r^{[0]}$, where ϵ^{i+1} $(0 < \epsilon < 1)$ is a decaying factor depending on the iteration count i, the extrapolated variables are adopted; otherwise, the variables before extrapolation are adopted. This strategy ensures convergence of the algorithm [22].

Algorithm 3 ADMM w/ an optimal step-size for Problem (3)

```
Input: $\mathbf{x}^{[0]}, \boldsymbol{\sigma}^{[0]}, \boldsymbol{\chi}^{[0]}, \boldsymbol{\xi}^{[0]}, \mathbf{z}^{[0]}, \boldsymbol{\eta}^{[0]}, \mathbf{y}_{\boldsymbol{\chi}}^{[0]}, \mathbf{y}_{\boldsymbol{\xi}}^{[0]}, \mathbf{y}_{\mathbf{z}}^{[0]}, \mathbf{y}_{\boldsymbol{\eta}}^{[0]}, \lambda > 0, p > 0,$
       $\boldsymbol{\zeta}_{\boldsymbol{\chi}} = p\boldsymbol{\chi}^{[0]} + \frac{1}{p}\mathbf{y}_{\boldsymbol{\chi}}^{[0]}, \boldsymbol{\zeta}_{\boldsymbol{\xi}} = p\boldsymbol{\xi}^{[0]} + \frac{1}{p}\mathbf{y}_{\boldsymbol{\xi}}^{[0]}, \boldsymbol{\zeta}_{\mathbf{z}} = p\mathbf{z}^{[0]} + \frac{1}{p}\mathbf{y}_{\mathbf{z}}^{[0]},$
       $\boldsymbol{\zeta}_{\boldsymbol{\eta}} = p\boldsymbol{\eta}^{[0]} + \frac{1}{p}\mathbf{y}_{\boldsymbol{\eta}}^{[0]}, \rho^{[0]} = p^2$
  for $i = 0, 1, 2, \ldots$ do
    $\mathbf{x}^{[i+1]} = \big((\boldsymbol{\chi}^{[i]} - \frac{1}{\rho^{[i]}}\mathbf{y}_{\boldsymbol{\chi}}^{[i]}) + (\boldsymbol{\xi}^{[i]} - \frac{1}{\rho^{[i]}}\mathbf{y}_{\boldsymbol{\xi}}^{[i]})\big)/2$
    $\boldsymbol{\sigma}^{[i+1]} = (\mathbf{B}^\mathsf{H}\mathbf{B} + \mathbf{I})^{-1}\big((\boldsymbol{\eta}^{[i]} - \frac{1}{\rho^{[i]}}\mathbf{y}_{\boldsymbol{\eta}}^{[i]}) + \mathbf{B}^\mathsf{H}(\mathbf{z}^{[i]} - \frac{1}{\rho^{[i]}}\mathbf{y}_{\mathbf{z}}^{[i]})\big)$
    $(\boldsymbol{\chi}^{[i+1]}, \boldsymbol{\eta}^{[i+1]}) = \mathrm{prox}_{\varphi/\rho^{[i]}}\big(\mathbf{x}^{[i+1]} + \frac{1}{\rho^{[i]}}\mathbf{y}_{\boldsymbol{\chi}}^{[i]}, \boldsymbol{\sigma}^{[i+1]} + \frac{1}{\rho^{[i]}}\mathbf{y}_{\boldsymbol{\eta}}^{[i]}\big)$
    $\boldsymbol{\xi}^{[i+1]} = P_{\mathcal{C}}\big(\mathbf{x}^{[i+1]} + \frac{1}{\rho^{[i]}}\mathbf{y}_{\boldsymbol{\xi}}^{[i]}\big)$
    $\mathbf{z}^{[i+1]} = \mathrm{prox}_{(\lambda/\rho^{[i]})\psi}\big(\mathbf{B}\boldsymbol{\sigma}^{[i+1]} + \frac{1}{\rho^{[i]}}\mathbf{y}_{\mathbf{z}}^{[i]}\big)$
    $\mathbf{y}_{\boldsymbol{\chi}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\chi}}^{[i]} + \rho^{[i]}(\mathbf{x}^{[i+1]} - \boldsymbol{\chi}^{[i+1]})$
    $\mathbf{y}_{\boldsymbol{\xi}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\xi}}^{[i]} + \rho^{[i]}(\mathbf{x}^{[i+1]} - \boldsymbol{\xi}^{[i+1]})$
    $\mathbf{y}_{\mathbf{z}}^{[i+1]} = \mathbf{y}_{\mathbf{z}}^{[i]} + \rho^{[i]}(\mathbf{B}\boldsymbol{\sigma}^{[i+1]} - \mathbf{z}^{[i+1]})$
    $\mathbf{y}_{\boldsymbol{\eta}}^{[i+1]} = \mathbf{y}_{\boldsymbol{\eta}}^{[i]} + \rho^{[i]}(\boldsymbol{\sigma}^{[i+1]} - \boldsymbol{\eta}^{[i+1]})$
    Compute the step-size $\rho^{[i+1]}$ (explained in Sec. IV-B)
  end for
```

B. Selecting Optimal Step-Size

The required number of iterations of ADMM depends on the parameter ρ, and its improper choice may lead to a slow algorithm. The recent study addresses the problem of selecting an optimal parameter by adaptively changing it at each iteration, improving the worst-case convergence rate of ADMM [23]. The ADMM algorithm with an optimal step-size is provided in Alg. 3, where the only difference from Alg. 1 is that the parameter ρ is not fixed but updated at each iteration as follows.

The optimal ρ is calculated using the following procedure [23]. First, let $\boldsymbol{\omega}^{[i]} = \left[(\boldsymbol{\chi}^{[i]})^\mathsf{T}, (\boldsymbol{\xi}^{[i]})^\mathsf{T}, (\mathbf{z}^{[i]})^\mathsf{T}, (\boldsymbol{\eta}^{[i]})^\mathsf{T}\right]^\mathsf{T}$, $\mathbf{y}^{[i]} = \left[(\mathbf{y}_{\boldsymbol{\chi}}^{[i]})^\mathsf{T}, (\mathbf{y}_{\boldsymbol{\xi}}^{[i]})^\mathsf{T}, (\mathbf{y}_{\mathbf{z}}^{[i]})^\mathsf{T}, (\mathbf{y}_{\boldsymbol{\eta}}^{[i]})^\mathsf{T}\right]^\mathsf{T}$, and $\boldsymbol{\zeta} = \left[\boldsymbol{\zeta}_{\boldsymbol{\chi}}^\mathsf{T}, \boldsymbol{\zeta}_{\boldsymbol{\xi}}^\mathsf{T}, \boldsymbol{\zeta}_{\mathbf{z}}^\mathsf{T}, \boldsymbol{\zeta}_{\boldsymbol{\eta}}^\mathsf{T}\right]^\mathsf{T}$, and find at most four roots of the following quartic equation with respect to p:

$$p^4\|\boldsymbol{\omega}^{[i+1]}\|_2^2 - p^3\langle\boldsymbol{\omega}^{[i+1]}, \boldsymbol{\zeta}\rangle_{\mathbb{R}} + p\,\langle\mathbf{y}^{[i+1]}, \boldsymbol{\zeta}\rangle_{\mathbb{R}} - \|\mathbf{y}^{[i+1]}\|_2^2 = 0, \qquad (13)$$

where $\langle\cdot,\cdot\rangle_{\mathbb{R}}$ denotes the real part of the inner product. Then, let $\mathcal{P}$ be the set of all real roots of Eq. (13), and select $p^\star$ by

$$p^\star \in \arg\min_{p \in \mathcal{P}} \|p\boldsymbol{\omega}^{[i+1]} + \mathbf{y}^{[i+1]}/p - \boldsymbol{\zeta}\|_2^2, \qquad (14)$$

i.e., select the real root that minimizes the above squared error. Using this $p^\star \in \mathbb{R}$, the step-size used in the next iteration is finally given by $\rho^{[i+1]} = (p^\star)^2$. Note that the roots of Eq. (13) can be obtained in the closed form, and selecting an optimal ρ preserves convergence of ADMM, see [23].

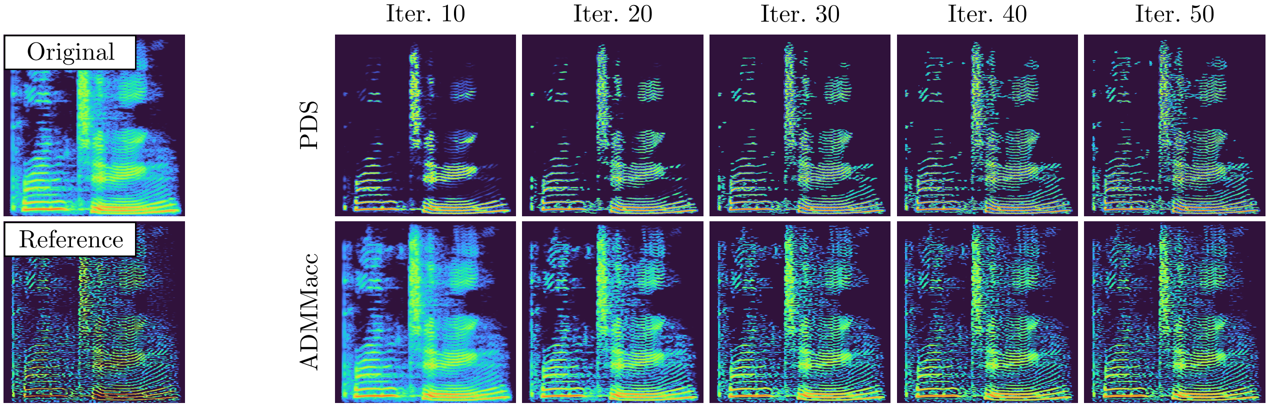

Fig. 1. Structured sparse T-F representations obtained by the conventional PDS and the proposed ADMMacc when using the function $\psi \circ \mathbf{B} = \|\mathbf{D}_{2\mathrm{D}}(\cdot)\|_{2,1}$ with $\lambda = 1/4$. The upper left (Original) is the T-F representation obtained by DGT and the lower left (Reference) is one obtained by ADMM ($\rho = 4.5$) after one million iterations. The upper and lower right show results from PDS ($\rho = 1.99$) and ADMMacc ($\rho = 10$), respectively, after 10, 20, 30, 40 and 50 iterations. All figures are illustrated with the color range of 100 dB.

V. Numerical Example

In this section, we compare the iteration counts for obtaining the structured sparse T-F representation of an acoustic signal. Let us refer to the ADMM algorithms with Nesterov acceleration and an optimal step-size as *ADMMacc* and *ADMMopt* hereinafter, respectively. We compared four algorithms for solving Problem (3): PDS [18], standard ADMM (Alg. 1), ADMMacc (Alg. 2), and ADMMopt (Alg. 3). Since Problem (3) cannot be solved analytically, we first obtained the T-F representations by running the standard ADMM ($\rho = 4.5$) over one million iterations and treated it as a reference. We then compared how quickly each method obtained a result close to this reference. In our experiments, we analyzed a speech signal with the sampling rate of 22 050 Hz using the Hann window of size $L = 2^9$ with hop size $a = 2^6$ and number of frequency bins $M = 2^{12}$. We used three different functions $\psi \circ \mathbf{B}$ to induce specific structures on the magnitude: $\|\cdot\|_*$, $\|\mathbf{D}_{2\mathrm{D}}(\cdot)\|_{2,1}$, and $\|\mathbf{F}\mathbf{D}_{2\mathrm{D}}(\cdot)\|_{2,1}$ with $\lambda = 25$, $1/4$, and $1/2$, respectively. The initial values for all algorithms were set to $\mathbf{x}^{[0]} = \mathbf{G}_{\mathrm{w}}\mathbf{s}$ and $\boldsymbol{\sigma}^{[0]} = \mathbf{1}$, and all the dual varibles were set to $\mathbf{0}$. In Alg. 1–3, the other variables were initialized with $\boldsymbol{\chi}^{[0]} = \boldsymbol{\xi}^{[0]} = \mathbf{x}^{[0]}$, $\mathbf{z}^{[0]} = \mathbf{B}\boldsymbol{\sigma}^{[0]}$, and $\boldsymbol{\eta}^{[0]} = \boldsymbol{\sigma}^{[0]}$. The decaying factor in Alg. 2 was set to $\epsilon = 0.95$ and the initial value of p in Alg. 3 was set to $p = 2$.

First, we conducted a visual comparison of the quickness of obtaining structured sparse T-F representations. The example of structured sparse T-F representations $|\mathbf{x}|$ obtained by PDS ($\rho = 1.99$) and ADMMacc ($\rho = 10$) is shown in Fig. 1. The proposed ADMMacc achieved a T-F representation close to the reference with fewer iterations than PDS, especially in the top half of the T-F representation.

Next, we quantitatively evaluated the difference among the algorithms. As an evaluation metric, we adopted the multi-scale structural similarity (MS-SSIM) [31] that quantifies image similarity based on luminance, constrast, and structure as it reflects similarity in terms of structural properties. MS-SSIM for each algorithm is illustrated in Fig. 2. As shown in the colored lines, the proposed ADMM algorithms achieved higher MS-SSIM with fewer iterations than the PDS algorithm, regardless of the choice of $\psi \circ \mathbf{B}$. In addition, ADMMacc (dashed lines) outperformed the standard ADMM (solid lines) for every step-size ρ, which confirms effectiveness of the acceleration. ADMMopt showed relatively high performance without selecting a step-size ρ, but it was not better than ADMMacc ($\rho = 10$). Constructing a method for obtaining the best step-size ρ is remained as a future work.

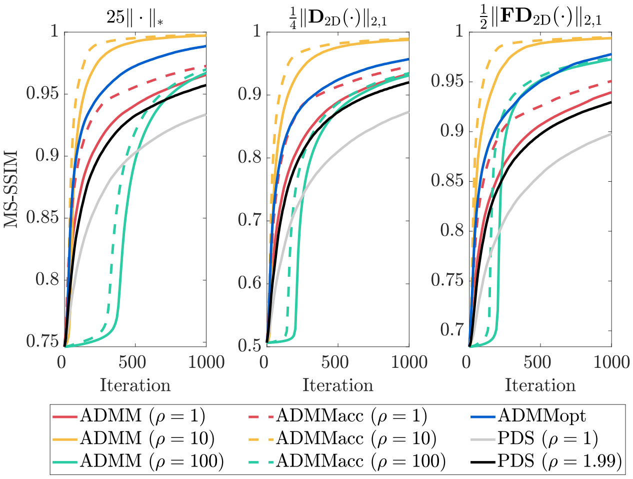

Fig. 2. The transition of MS-SSIM (every 10 iterations up to 1000) between the reference and the T-F representations obtained by each algorithm. The function $\lambda\psi \circ \mathbf{B}$ is shown above each figure.

VI. Conclusion

In this paper, we proposed the use of accelerated ADMM for realizing structured sparse T-F representations. We also provide specific examples of the penalty functions that can be computed efficiently. The experiments showed that the proposed algorithms can obtain structured sparse T-F representations with fewer iterations than the conventional PDS algorithm. Future works include investigation of computational time of the proposed algorithm as well as exploration of its specific application to acoustic signal processing.

References

[1] L. Cohen, *Time-frequency analysis*. Prentice Hall, New Jersey, 1995.

[2] K. Gröchenig, “Foundations of time-frequency analysis,” *Springer Sci. Bus. Media*, 2001.

[3] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[4] G. E. Pfander and H. Rauhut, “Sparsity in time-frequency representations,” *J. Fourier Anal. Appl.*, vol. 16, no. 2, pp. 233–260, 2010.

[5] P. Flandrin and P. Borgnat, “Time-frequency energy distributions meet compressed sensing,” *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 2974–2982, 2010.

[6] A. Gholami, “Sparse time–frequency decomposition and some applications,” *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 6, pp. 3598–3604, 2012.

[7] E. J. Candes, M. B. Wakin, and S. P. Boyd, “Enhancing sparsity by reweighted ℓ_1 minimization,” *J. Fourier Anal. Appl.*, vol. 14, pp. 877–905, 2008.

[8] Y. C. Eldar, P. Kuppinger, and H. Bölcskei, “Block-sparse signals: Uncertainty relations and efficient recovery,” *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3042–3054, 2010.

[9] R. Chartrand, “Shrinkage mappings and their induced penalty functions,” *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, pp. 1026–1029, 2014.

[10] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, “Structured sparsity through convex optimization,” *Stat. Sci.*, vol. 27, no. 4, pp. 450–468, 2012.

[11] M. Kowalski, K. Siedenburg, and M. Dörfler, “Social sparsity! neighborhood systems enrich structured shrinkage operators,” *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2498–2511, 2013.

[12] P. Yin, Y. Lou, Q. He, and J. Xin, “Minimization of $\ell_{1\text{-}2}$ for compressed sensing,” *SIAM J. Sci. Comput.*, vol. 37, no. 1, pp. A536–A563, 2015.

[13] M. Kowalski, A. Meynard, and H.-T. Wu, “Convex optimization approach to signals with fast varying instantaneous frequency,” *Appl. Comput. Harmon. Anal.*, vol. 44, no. 1, pp. 89–122, 2018.

[14] K. Tsubasa, K. Yatabe, and Y. Oikawa, “Sparse time-frequency representation via atomic norm minimization,” *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, pp. 5075–5079, 2021.

[15] H. Kuroda and D. Kitahara, “Block-sparse recovery with optimal block partition,” *IEEE Trans. Signal Process.*, vol. 70, pp. 1506–1520, 2022.

[16] F. J. Canadas-Quesada, P. Vera-Candeas, N. Ruiz-Reyes, J. Carabias-Orti, and P. Cabanas-Molero, “Percussive/harmonic sound separation by non-negative matrix factorization with smoothness/sparseness constraints,” *EURASIP J. Audio Speech Music Process.*, vol. 26, no. 1, pp. 1–17, 2014.

[17] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, “Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 24, no. 9, pp. 1626–1641, 2016.

[18] K. Arai, K. Yamada, and K. Yatabe, “Versatile time-frequency representations realized by convex penalty on magnitude spectrogram,” *IEEE Signal Process. Lett.*, vol. 30, pp. 1082–1086, 2023.

[19] L. Condat, D. Kitahara, A. Contreras, and A. Hirabayashi, “Proximal splitting algorithms for convex optimization: A tour of recent advances, with new twists,” *SIAM Rev.*, vol. 65, no. 2, pp. 375–435, 2023.

[20] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[21] T. Goldstein, B. O’Donoghue, S. Setzer, and R. Baraniuk, “Fast alternating direction optimization methods,” *SIAM J. Imaging Sci.*, vol. 7, no. 3, pp. 1588–1623, 2014.

[22] A. Buccini, P. Dell’Acqua, and M. Donatelli, “A general framework for ADMM acceleration,” *Numer. Algorithms*, vol. 85, pp. 829–848, 2020.

[23] Y. Ran, “General optimal step-size for ADMM-type algorithms: Domain parametrization and optimal rates,” *arXiv:2309.10124v2*, 2024.

[24] P. L. Combettes, “Perspective functions: Properties, constructions, and examples,” *Set-Valued Var. Anal.*, vol. 26, no. 2, pp. 247–264, 2018.

[25] P. L. Combettes and C. L. Müller, “Perspective maximum likelihood-type estimation via proximal decomposition,” *Electron. J. Stat.*, vol. 14, no. 1, pp. 207–238, 2020.

[26] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Phys. D: Nonlinear Phenom.*, vol. 60, no. 1-4, pp. 259–268, 1992.

[27] B. Recht, M. Fazel, and P. A. Parrilo, “Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization,” *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.

[28] K. Yatabe and K. Daichi, “Determined BSS based on time-frequency masking and its application to harmonic vector analysis,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 29, pp. 1609–1625, 2021.

[29] N. Parikh and S. Boyd, “Proximal algorithms,” *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, 2014.

[30] P. C. Hansen, J. G. Nagy, and D. P. O’leary, *Deblurring images: matrices, spectra, and filtering*. SIAM, 2006.

[31] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multiscale structural similarity for image quality assessment,” in *Proc. Thirty-Seventh Asilomar Conf. Signals Syst. Comput.*, vol. 2, 2003, pp. 1398–1402.